
IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re application of: Michael G. Martinek et al.

Attorney Docket No.: IGT1P369/SH00052-001

Application No.: 09/520,405

Examiner: Frank M. Leiva

Filed: March 8, 2000

Group: 3717

Title: COMPUTERIZED GAMING SYSTEM,
METHOD AND APPARATUS

Confirmation No.: 1300

CERTIFICATE OF EFS-WEB TRANSMISSION

I hereby certify that this correspondence is being transmitted electronically through EFS-WEB to the Commissioner for Patents, P.O. Box 1450 Alexandria, VA 22313-1450 on May 4, 2012.

Signed: /Michelle Heymann/
Michelle Heymann

PRE-APPEAL BRIEF REQUEST FOR REVIEW

Mail Stop AF
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Dear Madam:

Applicants respectfully request review of the final rejection in the above-identified patent application. No amendments are being filed with this request. This request is being filed with a Notice of Appeal.

Claims 58-70, 74-76, 78, 79, 81, and 83 are pending. Claims 58-70, 74-76, 78, 79, 81, and 83 were rejected under 35 U.S.C. §103(a) as being unpatentable over U.S. Patent No. 5,592,609 to Suzuki et al. (hereinafter “Suzuki”) in view of U.S. Patent No. 6,315,666 to Mastera et al. (hereinafter “Mastera”). Applicants respectfully traverse in view of the following.

Suzuki Does Not Disclose or Suggest a System Handler Application that Loads, Executes, and Unloads a Plurality of Gaming Program Shared Objects One at a Time During Execution of a Computerized Wagering Game

Claim 58 recites a “plurality of gaming program shared objects” that may “dynamically link” with a system handler application of an operating system “wherein the plurality gaming program

shared objects are functional units of game code that provide a particular feature set for the computerized wagering game, and wherein the system handler application loads, executes, and unloads the plurality of gaming program shared objects one at a time during execution of the computerized wagering game.”

As an initial matter, the Examiner’s rejection of claim 58 does not cite Suzuki for these features. Thus, it is respectfully submitted that Suzuki does not disclose or suggest these features, as recited in claim 58. However, the Examiner cites two portions of Suzuki in his rejection of claims 60 and 70, which allegedly recite features similar to the above-referenced features of claim 58. Because claims 60 and 70 are directed at subject matter having a different scope than claim 58, it is respectfully submitted that for at least these reasons, the rationale stated in the rejections of claim 60 and claim 70 is not applicable to claim 58. However, even if the rationale stated in the rejections of claim 60 and claim 70 was taken to be applicable to claim 58, neither of the referenced portions discloses or suggests the features recited in claim 58.

The first portion of Suzuki cited by the Examiner in the rejection of claim 60 describes access settings which allow a user to access a computer file. (Suzuki; Col. 28, lines 23 – 36). For example, “[t]he model software includes a base file and a user file. The base file is a file which a user cannot change and includes a portion of the model software and its associated original game which is not changeable by the user. The user file is a file which may be changed by the user. The user file includes default programs and default data which a user may modify to create user original programs and user original data.” (Suzuki; Col. 28, lines 23 – 29). While the referenced portion of Suzuki describes access parameters associated with computer files, the referenced portion makes no mention of an operating system or a system handler application included in an operating system. Thus, the referenced portion of Suzuki provides no guidance regarding the operation of a system handler application. Thus, the first portion of Suzuki referenced by the Examiner makes no disclosure or suggestion of a “system handler application [that] loads, executes, and unloads the plurality of gaming program shared objects one at a time during execution of the computerized wagering game,” as claimed.

The second portion of Suzuki cited by the Examiner in the rejection of claim 70 describes a central processing unit (CPU) and a pluggable random access memory (RAM) cartridge that may be included in a multiprocessor based game processor console. (Suzuki; Col. 2, line 60 – Col. 3, line 14). “The main CPU and game CPU may cooperate in the game execution and editorial process

such that an editing screen generated by the main CPU is superimposed on a game screen generated by the program executing the CPU.” (Suzuki; Col. 3, lines 5-10). As an initial matter, the referenced portion of Suzuki focuses on the operation of the CPU and makes no mention of the operating system. As one of ordinary skill in the art would appreciate, a CPU, which is a hardware processor, is not an operating system, which may be a program used to manage hardware. Therefore, while Suzuki describes a CPU and a pluggable RAM cartridge, Suzuki’s brief and vague discussion of the operation of a CPU provides no guidance regarding the operation of an operating system, or the operation of a system handler application included in the operating system. Thus, the second portion of Suzuki referenced by the Examiner makes no disclosure or suggestion of a “system handler application [that] loads, executes, and unloads the plurality of gaming program shared objects one at a time during execution of the computerized wagering game,” as claimed.

Furthermore, the Examiner may attempt to assert the mere disclosure of an operating system or a system handler application allegedly inherently discloses a “system handler application [that] loads, executes, and unloads the plurality of gaming program shared objects one at a time during execution of the computerized wagering game” in the claimed fashion. However, “in relying upon the theory of inherency, the examiner must provide a basis in fact and/or technical reasoning to reasonably support the determination that the allegedly inherent characteristic necessarily flows from the teachings of the applied prior art.” (MPEP 2112(IV)). “The mere fact that a certain thing may result from a given set of circumstances is not sufficient.” (MPEP 2112(IV)). As one of ordinary skill in the art would appreciate, it is possible that an operating system may load multiple data objects prior to execution of a game, execute the game, and subsequently unload the data objects, as opposed to loading, executing, and unloading the data objects “one at a time” during execution of the game, in the claimed fashion. Therefore, a “system handler application [that] loads, executes, and unloads the plurality of gaming program shared objects one at a time during execution of the computerized wagering game” in the claimed fashion does not necessarily flow from the teachings of Suzuki. (See MPEP 2112(IV)).

Mastera Does Not Disclose a System Handler Application that Loads, Executes, and Unloads a Plurality of Gaming Program Shared Objects One at a Time During Execution of a Computerized Wagering Game

The Examiner fails to demonstrate how Mastera cures the deficiencies of Suzuki with respect to claim 58. The Examiner only addresses claim language reciting the loading of objects. For example, the Examiner states “loading occurring when necessary for a change in game operation.” (Office Action, page 4). While the Examiner describes the loading of data, the Examiner provides no indication of how the referenced portion of Mastera allegedly discloses executing and unloading a “plurality of gaming program shared objects one at a time during execution of the computerized wagering game,” in the claimed fashion. Thus, it is respectfully submitted that no part of the Examiner’s rejection of claim 58 directly addresses claim language reciting a “system handler application [that] loads, executes, and unloads the plurality of gaming program shared objects one at a time during execution of the computerized wagering game,” as claimed.

Moreover, even if the rejection were not deficient with respect to claim 58, Mastera does not disclose the features recited in claim 58. Mastera describes gaming machines having a secondary display for providing video content. The system disclosed in Mastera may further comprise a graphics controller and video RAM storing video data corresponding to displayed video content. The language cited by the Examiner describes a system controller determining when video data should be provided to a graphics controller. (Mastera; Col. 13, lines 10-13). For example, “when the system powers up or there is a change in the bonus game being played, all necessary audio and visual information for the new animations or early display is loaded into DRAM 619.” (Mastera; Col. 13, lines 5-8). While Mastera describes loading data into DRAM, Mastera makes no disclosure regarding the subsequent execution and unloading of the data, or whether gaming program shared objects are loaded, executed, and unloaded “one at a time,” in the claimed fashion. Accordingly, the referenced portion of Mastera makes no disclosure of a “system handler application [that] loads, executes, and unloads the plurality of gaming program shared objects one at a time during execution of the computerized wagering game,” as claimed.

Furthermore, the Examiner may attempt to assert that loading video data allegedly inherently discloses a “system handler application [that] loads, executes, and unloads the plurality of gaming program shared objects one at a time during execution of the computerized wagering game” in the claimed fashion. As discussed above, “in relying upon the theory of inherency, the examiner must

provide a basis in fact and/or technical reasoning to reasonably support the determination that the allegedly inherent characteristic necessarily flows from the teachings of the applied prior art.” (MPEP 2112(IV)). “The mere fact that a certain thing may result from a given set of circumstances is not sufficient.” (MPEP 2112(IV)). Loading data does not describe subsequent execution or unloading of the data. Moreover, the referenced portion of Mastera makes no disclosure of unloading data while the wagering game is executing. Because Mastera is silent regarding aspects of loading, executing, and unloading the data, it is possible that multiple data objects are loaded at once and subsequently executed, and not loaded, executed, and unloaded “one at a time” in the claimed fashion. Therefore, a “system handler application [that] loads, executes, and unloads the plurality of gaming program shared objects one at a time during execution of the computerized wagering game,” as claimed, does not necessarily flow from the teachings of Mastera. (See MPEP 2112(IV)).

The Combination of References Does Not Disclose or Suggest a System Handler Application that Loads, Executes, and Unloads a Plurality of Gaming Program Shared Objects One at a Time During Execution of a Computerized Wagering Game

Claim 58 recites a system handler application of an operating system “wherein the plurality gaming program shared objects are functional units of game code that provide a particular feature set for the computerized wagering game, and wherein the system handler application loads, executes, and unloads the plurality of gaming program shared objects one at a time during execution of the computerized wagering game.” For at least the reasons stated above, Suzuki and Mastera, when considered alone or in combination, fail to disclose or suggest the features recited in claim 58.

Claim 76 recites features similar to those recited in claim 58. It is respectfully submitted that claim 76 is patentable for at least similar reasons as claim 58. The dependent claims include, by virtue of their dependency, the features of the independent claims on which they are based and, therefore, are patentable for at least similar reasons. Therefore, it is respectfully submitted that the rejections to all pending claims should be withdrawn.

Weaver Austin Villeneuve & Sampson LLP
P.O. Box 70250
Oakland, CA 94612-0250
(510) 663-1100

Respectfully submitted,
/William J. Egan, III/
William J. Egan, III
Reg. No. 28,411